Long Range Outlook Data Transfer
Operations Guide

February 22, 2016

Office of Central Processing Hydrologic Support Group (OCP-Hydro) and
National Water Center (NWC-Silver Spring)

1.0 Introduction

This document describes the data transfer operations for the Long Range Outlook (LRO) map-based information posted on the NWS web pages (http://water.weather.gov/ahps/long_range.php). The LRO flood risk displays give probabilities of the threat of flooding over a three-month future time period for selected locations in the NWS hydrologic service area. The LRO information is presented on a map, and given as probabilities of exceeding various threshold river levels.

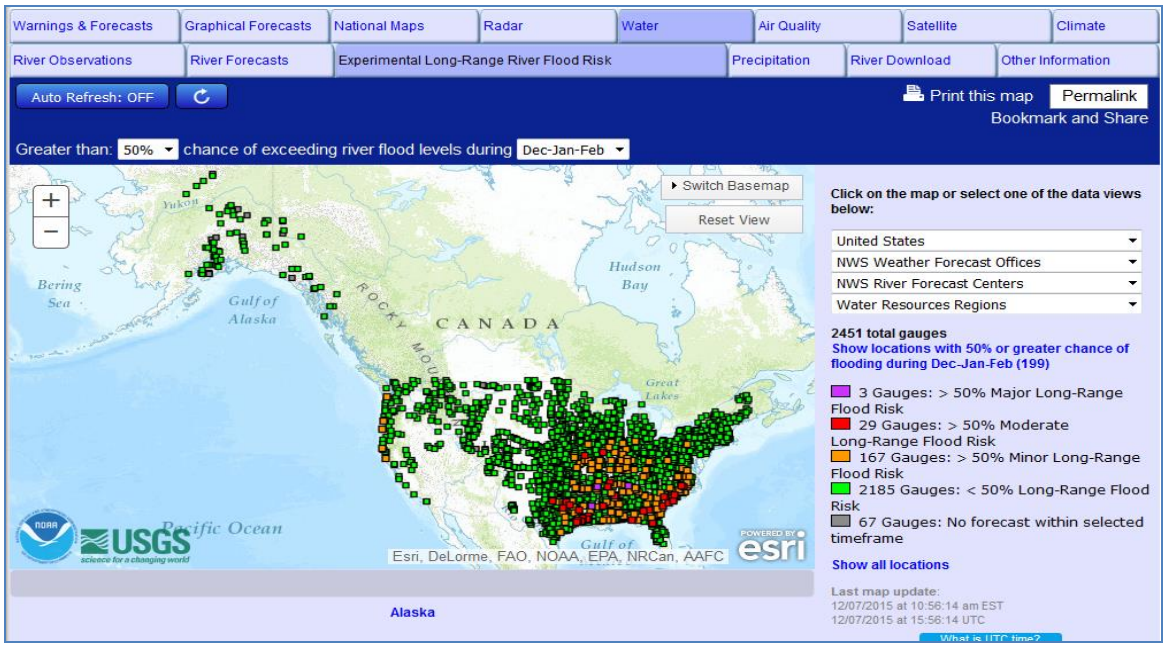

Figure 1. Sample Long Range Outlook Web Image

The data transfer function is part of an overall process for producing the web images and has the following three functional, sequential steps:

1) Generation: the Community Hydrologic Prediction System (CHPS) GraphGen application produces the data files containing raw LRO information. This function is performed at RFCs.

2) Transfer: parses this raw information, then analyzes and formats and sends the derived information, as noted below in the three sub-steps. This function is performed at RFCs.

3) Display: produces and hosts the web content displays from the transferred derived information. This function is performed at servers hosted by the NWS Information Dissemination Services (NIDS) and is not located at the RFC.

The data transfer described as step 2 above involves the following processing steps:

a) Analysis and Formatting of the information from the 3 stage threshold (minor, moderate, major) files produced by GraphGen into a data file containing processed LRO display-ready data.

b) Forwarding the data file to the AWIPS Local Data Acquisition and Dissemination (LDAD) server.

c) Sending of the file from LDAD to the external NIDS functions which host the web services. LDAD must be used because it is the only system with network access to NIDS.

This document focuses on this data transfer function and also explains how to setup the file generation using CHPS GraphGen. The display function is only briefly covered in this document.

2.0 LRO Overview

The generation of the LRO information disseminated to the web services is performed by the River Forecast Centers (RFCs). Other hydrologic web displays, that are accessible via water.weather.gov, display observational and deterministic forecast data provided by the Weather Forecast Offices (WFO), and probabilistic graphs provided by the RFCs.

This latter probabilistic information is presented in the form of two graphs:

- "Weekly Chance of Exceeding Levels" graph
- "Chance of Exceeding Levels During Entire Period" graph

Like the colored, annotated icons shown on the LRO maps, as in Figure 1, both of these exceedance graphs are for 3-month periods. While the process for generating these exceedance graphs is not described herein, the information on these graphs is inherently related to the LRO map-based information.

It is very important that the map-based LRO information and the more detailed graphical probabilistic information are numerically consistent. It is also important to have the graph-based and map-based information represent as close to the same time period as possible. This requires coordination of the processes for generating the two complementary sets of information. The two processes are separate however, so careful sequencing and scheduling of operations is important to achieve this consistency.

2.1 LRO Transmission Timing

The timing of the LRO operations has to be carefully managed to ensure consistent and meaningful displays. The monthly timing for the full annual cycle is shown in the table below. The generation and transmission of the LRO information must be performed at least once a month, before or on the 28th day of the month that precedes the subsequent 3-month period.

Table 1. Timing for NIDS Use of 3-Month Products

For 3-month forecast period	Provide grids NLT 23z on	Updates accepted until 23Z on
Jan-Feb-Mar	December 28	January 28
Feb-Mar-Apr	January 28	February 28
Mar-Apr-May	February 28	March 28
Apr-May-Jun	March 28	April 28
May-Jun-Jul	April 28	May 29
Jun-Jul-Aug	May 28	June 28
Jul-Aug-Sep	June 28	July 28
Aug-Sep-Oct	July 28	August 28
Sep-Oct-Nov	August 28	September 28
Oct-Nov-Dec	September 28	October 28
Nov-Dec-Jan	October 28	November 28
Dec-Jan–Feb	November 28	December 28

This timing is critical because the NIDS web page starts showing the new 3-month period on the 28th at 23z of the preceding month. Updates can be provided until the 28th of the first month of the 3-month period. There is no limit on the number of updates you can send. As a rule, the last update for a given period is used and the previous updates are overwritten. The NIDS process currently checks for updates once an hour.

2.2 LRO Information Usage

Information for a given 3-month period can be provided up to a 11 months earlier than a 3-month period cutover date without any consequence to the currently displayed 3-month period. Information sent early will not be used until the cutover date for the period, per the cycle dates. For example, the data for Jun-Jul-Aug 2016 can be sent any time after June 28, 2015 at 23:00z, with the data being used on May 28, 2016 23:00Z. The most accurate outlook information is naturally best delivered just before the 3-month cutover, so sending datasets too early is not recommended.

Datasets are uniquely identified for each 3-month period, using the first letter in the name of the three months (i.e. October-November-December is “OND”). In this manner, the files do not overwrite each other and NIDS retains data for each of the (12) unique 3-month periods.

Information is always managed in 3-month periods, and the same 3-month period is applied to all RFCs. Once the cutover to a new 3-month period occurs on the 28th, then even if the first two months of the new period are covered by previously sent 3-month dataset, that information is ignored and not considered at all. As an example, if the March-April-May information is provided in late February, but nothing is provided after that, then on March 28, the NIDS map will show missing information and make no attempt to use the April-May information for the first two months of the April-May-June period.

2.3 LRO Station Definition

A station shown on the LRO maps must be defined in the AHPS Content Management System (CMS) (https://nwscms.nids.noaa.gov) as a probabilistic station by having the "Probabilistic Site" dropdown on the "Edit Gauge" page set to "Yes". A probabilistic station will have a map station icon displayed on the LRO map displays.

This switch setting also controls the display of the "Chance of Exceedance" graphics for individual stations. If the switch is turned off, then the station is not displayed on the LRO map pages nor is the station icon presented in either the Observed or Forecast map displays as a probabilistic station.

3.0 Methods for RFC Generation and Transmission

The RFCs are responsible for executing the first two of the three overall steps that result in the LRO web displays (i.e. generation and transfer).

The first of these two steps is the production of the 3 stage threshold "table" files (LRO.minor.tbl, LRO.moderate.tbl, LRO.major.tbl) that contain the LRO probabilistic information, using the CHPS GraphGen application. A GraphGen product template file is provided as part of the baseline software distribution. This product template file has important instructions on properly defining the time period being considered, which is dependent in part on the current date.

NCRFC provided the following comments on the GraphGen setup:

- GraphGen uses the thresholds defined in ThresholdValueSets via the LRO GraphGen templates: inputSeriesProviderParameters includeThresholds="true" name "ConfiguredQuery" and the parameter id queryId of SSTG. This "SSTG" should work for most RFCs if they have SSTG set up correctly in their PiServiceConfigFile.

- The GraphGen PiServiceConfigFile has a timeSeries id called SSTG... and each RFC will have that set up a little differently as far as how the timeSeriesSet looks... but it uses a moduleInstanceSet & locationSet of SSTG timeSeries that *should* have the various ThresholdValueSets defined.

The second of these two steps is the processing of the resulting GraphGen files into a single file ultimately sent to NIDS for use in the web display process. The steps involved in this "transfer" process are given below with reference to the software components performing the function. These components are described later.

a) Analysis and formatting parses the information in the 3 stage threshold table files (LRO.minor.tbl, LRO.moderate.tbl, LRO.major.tbl) and generates a single file (*xx*RFC.LRO.*mmm*, where *xx* is the RFC ID and *mmm* is the first letter of each month) with all the LRO information for each flood threshold. These actions are performed by a Python script (LRO.py).

b) Forward the LRO output file to LDAD for use in the subsequent transmission step. (typically done by LRO.py script). NOTE: At the end of the LRO.py script is a commented-out os.system call that

secure copies the output file to ls1. Each RFC will have to adapt this command to send the output file as they see fit.

c) Send the LRO output file to NIDS. This is done from LDAD servers using rsync commands contained within a shell script (typically done by a local script). Note: A script to perform this step is not provided as part of the CHPS baseline package.

These three steps are performed using files and methods described in the next section. Some of these files are provided with the CHPS baseline software distribution.

4.0 LRO Software Distribution

The original LRO Formatter scripts, files and documentation the NWC started with to create its CHPS version of the LRO Formatter were retrieved from the NWS Software Collaboration Portal (SCP) wiki page at:
https://collaborate.nws.noaa.gov/trac/nwsscp/wiki/HydroRfc/LroFormatter

As of release CHPS 5.3.1/OHD-CORE-CHPS-4.4.a, the LRO Formatter has been added to the CHPS baseline.

Note: With the CHPS implementation of the LRO Formatter

1. The collection_data script is no longer needed and will not be delivered.
2. Minor changes have been made to the original LRO.py script to pass in two arguments to set the RFC ID and define the parent directory where its input and output directories are placed.

A description for the files used throughout the LRO Formatter process is in the table below:

File(s)	Purpose	Notes
global.properties	Define the location of the LRO python script and where its input and output data files are placed.	Need to add the following two properties: **OHD_SCRIPTS_DIR** – Location of the LRO.py script. **LRO_DIR –** Parent directory for LRO ./input & ./output. See Appendix A for setup instructions.
LRO.xml	This is the GraphGen product template file definition for LRO.	This file is located in the CHPS OHD-CORE package under ./graphgen/installFiles/graphgen/LRO/ This LRO product template file is used as input to GraphGen, which produces the LRO.*.tbl files. The contents of this file needs to be included in the GraphGen Product Template file (OHD_PRODUCTS_AND_SETTINGS.xml) and can be manually entered or imported using the GraphGen GUI. See Appendix A for setup instructions.

GraphGen_AHPS_Products_LRO.xml Note: Your file may have a different name. This is up to each RFC to define.	This module file generates the 3 stage threshold "table" files for all segments and creates the *xx*RFC.LRO.*mmm* file.	This file is located under ./moduleConfigFiles/graphgen/ This is a LRO module General Adapter file which contains the instructions to: 1. Set the directory where the *.tbl files are placed. 2. Purge the previous *.tbl files found in the LRO input directory. 3. Define the products for all segment ids. 4. Execute GraphGen to produce the *.tbl files. 5. Execute the LRO.py script with the two required arguments. See Appendix A for setup instructions.
LRO.major.tbl, LRO.moderate.tbl, LRO.minor.tbl	Input files for the LRO Formatter script.	These files are located at $LRO_DIR/input/ These three files are created by GraphGen and processed by LRO.py. They contain semi-formatted probabilistic information for each location defined in the LRO module General Adapter file.
LRO.py	LRO Formatter script.	This file is located at $OHD_SCRIPTS_DIR/ This script executed from the LRO module General Adapter file, parses the three .tbl files, and creates an output file (*xx*RFC.LRO.*mmm*) for transmission to NIDS. It also creates a LRO.log file.
LRO.log	Logging file for the LRO Formatter script.	This file is located at $LRO_DIR/output/ This file contains information about the execution of the LRO.py script. It is used for debugging purposes.
xxRFC.LRO.mmm	File used by NIDS to display LRO on the web.	This file is located at $LRO_DIR/output/ This file contains the processed LRO display-ready data for the 3-month period *mmm* which can be forwarded to the AWIPS LDAD and then sent on to NIDS.

An example of a LRO module General Adapter file (GraphGen_AHPS_Products_LRO.xml) for NCRFC is provided with the CHPS OHD-CORE release package and is found under the directory ./graphgen/installFiles/graphgen/LRO/ModuleConfigFiles/graphgen/

It is recommended that you verify your AHPS LRO product configuration using a FEWS Stand-Alone (SA) before configuring it for your Operator-Client (OC). See Appendix B for how to verify your AHPS LRO product configuration within a SA.

A workflow should be scheduled through the CHPS/FEWS AI Scheduler to run the LRO module General Adapter file (ie. GraphGen_AHPS_Products_LRO.xml) every few days to produce stage threshold table files (*.tbl) with the most current data available.

The CHPS Release package does not include any baseline cron scripts that perform the forwarding of the xxRFC.LRO.mmm file to LDAD or the final step of sending the file to NIDS.

4.1 Analysis and Formatting Files

The LRO.py script performs the core processing described in this document. Ultimately, it creates a file for each categorical threshold that is sent to NIDS. The file created by the LRO script and sent from the RFCs to NIDS contains the following information:

- Id – location identifier
- *fgroup – forecast group*
- *city,name,stream,hsa – location attributes*
- *fs, wstg, minor_stage, moderate_stage, major_stage –threshold stages*
- *lat, lon – latitude/longitude*
- *stntype – station type*
- minor_CS_3month, moderate_CS_3month, major_CS_3month – threshold conditional simulation probabilities
- date_produced – generation date
- obj_or_subj_3month – objective or subjective forecast indicator
- interval_3month – three letter code for three-month period

By default, the LRO script assigns “obj”, and does not assign values to the shaded fields listed between fgroup and stntype. NIDS obtains the latitude/longitude from the CMS information for each station.

4.2 Forwarding Files to LDAD

As noted earlier, this action is commented out in the LRO.py script because the target directory on LDAD may be uniquely defined at each RFC.

Resulting output must be configured to be sent to LDAD server (ls1) using scp command via a crontab.

```
scp <$LRO_DIR/output/XXRFC.LRO.mmm>
ldad@ls1:/data/ldad/public/images/xxrfc/data/esp_natlmap'
```

Where XX and xx = two-character RFC identifier
mmm=three-character month designation (e.g. DJF=December/January/February)

Example:

```
scp
</awips/chps_share/oc/fews/ncrfc_oc/Export/GraphGen/LRO/output/NCRFC.LRO.DJF>
ldad@ls1:/data/ldad/public/images/ncrfc/data/esp_natlmap'
```

4.3 Sending Files to NIDS

The information is sent to NIDS using "rsync" (remote sync) processes. Note that the system performing this rsync must have its IP address previously registered on an access list for the sysnc.weather.gov server. NIDS must be contacted if there are any changes needed for this access list.

For background, the steps for the transmission process are summarized below.

1) Transmission Host: The AWIPS system from which the LRO data files will be rsynced must be the same system which has the external IP address that was provided for the sync of the Probability of Exceedance (POE) graphics also sent to NIDS.

2) Transmission Execution: This is executed via the below command.

   Replace `{path_to_data_file_directory}` with the proper path

   ```
   /usr/bin/rsync -ltDvzh --progress --timeout=60
   {path_to_data_file_directory}/* sync.weather.gov::ahps_long_range
   ```

   Example:

   ```
   /usr/bin/rsync -ltDvzh --progress --timeout=60
   /home/ahps/long_range_data/* sync.weather.gov::ahps_long_range
   ```

   Only the LRO data files should be sent. No directories or other files should be sent. Please ensure the individual LRO data files have the permissions set as "-rw-rw-r--" before sending, as permissions other than these may cause the data files to not be parsable to the AHPS NIDS processes.

3) Transmission Scheduling: This send process can be either executed manually via Step #2 or it can be automated via a crontab on LDAD with a comment of the form:

   ```
   0 * * * * ldad /usr/bin/rsync -ltDvzh --progress --timeout=60
   {path_to_data_file_directory}/* sync.weather.gov::ahps_long_range >
   /tmp/rsync_LRO_files.log 2>&1
   ```

   The command execution frequency via the crontab should be customized based upon individual RFC practices. The above crontab example syncs the Long Range Flood Risk data files each hour. Other custom approaches may be used. For example, it would be helpful to have the log file retained for some modest period of time. The log file output is critical and must be maintained for troubleshooting purposes.

Once the graphics are transmitted to sync.weather.gov, they are automatically disseminated to the NIDS web servers, typically every 15 minutes. The LRO files on both MO and MD are parsed and inserted/updated into to the NIDS database every hour, on the hour. The web-accessible LRO ESRI map

queries the database every time the page loads; therefore, it will always pull exactly what is in the database and shouldn't change until the next hour. There is no pre-generated imagery for LRO, all images are assembled dynamically straight out of the database.

The downloadable LRO CSV file on both MO and MD are generated every 15 minutes. Data may also be sent to NIDS developmental environment every 15 minutes.

5.0 NIDS Display Operations

There are two NIDS servers that receive the RFC files. The files are generally of the form:

http://water-ss.weather.gov/resources/rsync/long_range/rrRFC.LRO.mmm

- ss=two-character NIDS server identifier (character
- rr= two-character RFC identifier
- mmm=three-character month designation (e.g. DJF=December/January/February)

The most direct way to monitor the LRO information display is to review the information on the web page itself. Grey stations icons indicate missing data.

You can confirm your data made it to the NIDS by checking the AHPS "River Download" tab. This displays a list of downloadable data sets, at http://water.weather.gov/ahps/download.php. Scroll down to the section with "Long Range" products. Select the "CSV" format listing for download.

All RFC data are compiled in this download file in comma-separated-value (CSV) format, updated every 15 minutes. This listing contains information for each reporting long range outlook station, including the "date produced" field and other fields provided by the RFCs, as detailed later. This download shows the data for the most recent period, which could be for a future three-month period.

NIDS retains the most recent file from each RFC, for each 3-month period, but it does not manage an archive of these previous year's 3-month files. It also does not provide a public interface to the prior month's 3-month files; the download feature mentioned above only provides the data delivered for the most recent period.

6.0 Reporting Issues with LRO Display

Errors associated with the LRO display include missing or incorrect data. The can be caused by issues in the functions that comprise the end-to-end process: generation, transfer, or display of the data.

Should errors be encountered with the connection or transmission to sync.weather.gov, a TOC ticket may be opened by detailing the issue and emailing: toc.nwstg@noaa.gov

Appendix A: Setting up the LRO Formatter in CHPS

1. Edit the global.properties file to define the location for the LRO python script and where its input and output data files are placed.

   OHD_SCRIPTS_DIR=/*<OHD-CORE-CHPS>*/**ohd/scripts**
   This is the Location for where the LRO.py script resides. *<OHD-CORE-CHPS>* needs to be the base directory where you install the OHD-CORE package on your system.

   LRO_DIR=/<path-to-input-output>/**LRO**
   This is the Location under which the LRO ./input & ./output directories reside. <path-to-input-output> is the base directory where the user would like the input and output directories for LRO Formatter to reside. Possibly the ./Export/GraphGen/LRO/ directory under your OC.

2. Verify that you have LRO defined in the GraphGen product template file (**OHD_PRODUCTS_AND_SETTINGS.xml**) similar to what is provided in the **LRO.xml** file delivered in the CHPS OHD-CORE package under **./graphgen/installFiles/graphgen/LRO/.** This can be done either by manually editing the file **OHD_PRODUCTS_AND_SETTINGS.xml** directly or importing the LRO.xml file using the GraphGen GUI.

3. NOTE: To simplify the support and maintenance for the CHPS implementation of the LRO Formatter, there will not be a collection_data script used, and therefore **only one single LRO module General Adapter should be set up for the entire RFC.** This single General Adapter will produce just one LRO.major.tbl, LRO.moderate.tbl, and LRO.minor.tbl table file for the entire RFC. Please verify that your LRO module General Adapter file contains the instructions to:
   a. Purge the previous *.tbl files found in the LRO input directory.
   b. Set the directory where the *.tbl files are placed.
   c. Define the products for all segment ids.
   d. Execute GraphGen to produce the *.tbl files.
   e. Execute the LRO.py script.

   Below is an example of a LRO module General Adapter (./ModuleConfigFiles/graphgen/GraphGen_AHPS_Products_LRO.xml) which has the five instructions mentioned above.

Note: The lines highlighted in green in this example file should not be changed and the lines highlighted in turquoise are to be modified.

```
<?xml version="1.0" encoding="UTF-8"?>
<generalAdapterRun xmlns="http://www.wldelft.nl/fews" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.wldelft.nl/fews http://chps1/schemas/generalAdapterRun.xsd">
 <general>
  <description>LRO GraphGen Products</description>
  <rootDir>%TEMP_DIR%</rootDir>
  <workDir>%ROOT_DIR%/work</workDir>
  <exportDir>%ROOT_DIR%/input</exportDir>
  <exportDataSetDir>%ROOT_DIR%</exportDataSetDir>
  <importDir>%ROOT_DIR%/output</importDir>
  <dumpFileDir>$GA_DUMPFILEDIR$</dumpFileDir>
```

```
  <dumpDir>%ROOT_DIR%</dumpDir>
  <diagnosticFile>%ROOT_DIR%/output/diag.xml</diagnosticFile>
 </general>
 <activities>
  <startUpActivities>
   <purgeActivity>
    <filter>$LRO_DIR$/input/*.tbl</filter>
   </purgeActivity>
   <makeDir>
    <dir>%ROOT_DIR%/output</dir>
   </makeDir>
   <makeDir>
    <dir>%ROOT_DIR%/work</dir>
   </makeDir>
   <makeDir>
    <dir>$EXPORT_DIR$/products</dir>
   </makeDir>
  </startUpActivities>
  <exportActivities>
   <exportRunFileActivity>
    <exportFile>%ROOT_DIR%/run_info.xml</exportFile>
    <properties>
     <int key="printDebugInfo" value="0"/>
     <string key="piServiceBackendRFCIdentifier" value="$piServiceBackendRFCIdentifier$"/>
     <string key="piServiceHostName" value="$piServiceHostName$"/>
     <string key="piServicePortNumber" value="$piServicePortNumber$"/>
     <string key="baseOutputDir" value="$LRO_DIR$/input"/>
     <string key="model" value="ohd.hseb.graphgen.adapter.GraphGenModelAdapter"/>
     <string key="ohdGraphgenCentralDir" value="$ohdGraphgenCentralDir$"/>

     <!-- Add all segment Ids here---------------------------------------- -->
     <string key="products.<SegmentId>" value="AHPS_LRO.*"/>

    </properties>
   </exportRunFileActivity>
  </exportActivities>
  <executeActivities>

   <!-- Generate LRO table files -->
   <executeActivity>
    <command>
     <className>ohd.hseb.graphgen.adapter.GraphGenModelAdapter</className>
     <binDir>$OHDBINDIR$</binDir>
    </command>
    <arguments>
     <argument>%ROOT_DIR%/run_info.xml</argument>
    </arguments>
    <!—Increase timeOut if more than three-hundred products.<SegmentId> are defined.
        20min = 1200000 -->
    <timeOut>300000</timeOut>
   </executeActivity>

   <!-- Analyzes and Formats the information -->
```

```
    <executeActivity>
     <command>
      <executable>$OHD_SCRIPTS_DIR$/LRO.py</executable>
     </command>
     <arguments>
      <argument>NCRFC</argument>
      <argument>$LRO_DIR$</argument>
     </arguments>
     <timeOut>300000</timeOut>
    </executeActivity>
   </executeActivities>
  </activities>
</generalAdapterRun>
```

An example of a LRO module General Adapter file (GraphGen_AHPS_Products_LRO.xml) for NCRFC is provided with the CHPS OHD-CORE release package and is found under the directory ./graphgen/installFiles/graphgen/LRO/ModuleConfigFiles/graphgen/

Appendix B: Verify AHPS LRO product configuration within a SA

After you have set up LRO in CHPS as described in Appendix A, it is suggested that you verify that your AHPS LRO product configuration is setup correctly using a FEWS Stand-Alone (SA). A modification to the LRO module General Adapter file needs to be made for this test. The steps are as follows:

1. Comment out the lines that define the piService* run file properties in the <SA_dir>/Config/moduleConfigFiles/graphgen/GraphGen_AHPS_Products_LRO.xml

   See the lines highlighted in green below:

```
<exportRunFileActivity>
<exportFile>%ROOT_DIR%/run_info.xml</exportFile>
<properties>
<!--
<string key=
"piServiceBackendRFCIdentifier" value="$piServiceBackendRFCIdentifier$"/>
<string key="piServiceHostName" value="$piServiceHostName$"/>
<string key="piServicePortNumber" value="$piServicePortNumber$"/>
-->
...
</properties>
</exportRunFileActivity
```

2. Start CHPS using the ./ohdplugins/fews_ohdPlugins.sh

3. Set the FEWS PI service port number by entering the number shown in the CHPS Logs Panel "Started FewsPiServiceImpl on localhost: 8100"

4. Execute the LRO products workflow via CHPS interface **Manual Forecast** dialog.

5. If executed successfully, a LRO product file (xxRFC.LRO.mmm) will be created in the output directory found under the LRO directory specified by the $LRO_DIR variable in the global property file. A log file (LRO.log) containing information about the execution of the LRO.py script will be in the output directory as well and can be used for debugging purposes.

6. Remember to uncomment the lines for the piService* run file properties in the GraphGen_AHPS_Products_LRO.xml before running in an OC.

Appendix C: Running the CHPS LRO Formatter in an OC

After you have set up LRO in CHPS as described in Appendix A, and tested it in a SA as described in Appendix B, you are now ready to upload these configurations in an OC and schedule the LRO workflow task via the FEWS AI.

Executing this workflow will result in an xxRFC.LRO.mmm file being created in the $LRO_DIR/output/ directory.

To get this file from the output directory to a place where it will be processed by NIDS, please refer to section 4.2 **Forwarding Files to LDAD** and section 4.3 **Sending Files to NIDS** of this document.

The following Appendix is for internal NWSHQ purposes to keep track of suggested requests for the LRO display on NIDS. RFC can ignore this information below.

Appendix D: Suggestions for Web Displays

D.1 "River Download" Report Features:

(a) Modify report to shows only the stations defined as probabilistic stations in the CMS, rather than any station that at one point delivered data. Currently over a hundred orphaned stations are listed, which makes it difficult to distinguish truly missing stations from retired stations. (High priority)

(b) Modify report to list the "date produced" date for the data that is actually used in the map display. Currently the report gives information for the last transmitted data, even if it is for a future period. (High priority)

(c) Consider adding a new LRO Ingest report as part of the AHPS toolbox at (http://water.weather.gov/monitor/tools.php). This report would be similar to the current river download report. Instead of showing the data for the current three-month period, it would summarize the information for each of the twelve three-month periods. This would be for solely for diagnostic monitoring and be helpful to show what data was actually sent versus what data is displayed. (Medium priority)

(d) Add a second new report or modify the new toolbox report mentioned above to monitor the consistency of the exceedance graphs with reference to the LRO information. This could include reports of the time the currently displayed graphs were sent and the time the currently display LRO was sent. If there is any way possible to show the precise three-month period represented by each graph and the LRO data, that would be ideal. Note that three-month periods can actually have a start date on any day of the month. (High Priority)

D.2 Main Interface:

(a) The three month LRO period is shown as an option menu. Given that there is no way to use this option menu to select the three-month period, disable the dropdown for the three-month period (e.g. Nov-Dec-Jan) or display it as an uneditable text box. (Low priority)

END